3-18-2020

Customized Wireless Mesh Routing Metric for Swarm of Drones Applications

Oscar G. Bautista Chia
Florida International University, obaut004@fiu.edu

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

CUSTOMIZED WIRELESS MESH ROUTING METRIC FOR SWARM OF DRONES APPLICATIONS

A thesis submitted in partial fulfillment of

the requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

by

Oscar Bautista Chia

2020

To: Dean John L. Volakis
College of Engineering and Computing

This thesis, written by Oscar Bautista Chia, and entitled Customized Wireless Mesh Routing Metric for Swarm of Drones Applications, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Selcuk Uluagac

Alexander Perez-Pons

Mohammad Rahman

Kemal Akkaya, Major Professor

Date of Defense: March 18, 2020

The thesis of Oscar Bautista Chia is approved.

Dean John L. Volakis
College of Engineering and Computing

Andres G. Gil
Vice President for Research and Economic Development and
Dean of the University Graduate School

Florida International University, 2020

DEDICATION

To my parents Gustavo and Rosa:

Without your example and sacrifice for several years, I would not be here undertaking this endeavor.

Father, thanks for always trusting me and supporting me in my aspirations and efforts. There will never be enough thanks.

To my wife and children:

I strive to work on what I like so I can do better. This endeavor is also for you.

Dreaming big is never wrong.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to Dr. Akkaya for his support and guidance throughout this research. Thanks Dr. Akkaya, for showing me where to improve and get this far as a researcher and as a professional.

I would like to extend my sincere thanks to Dr. Uluagac, for giving me the opportunity along with Dr. Akkaya to join your laboratory and work on a research project that became this thesis.

Thanks also to Dr. Saputro and in general all the lab members and friends who helped along the way with useful tips, tools, references, examples and answers that aided in advancing this work.

This publication was made possible by NPRP grant #NPRP9-257-1-056 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

ABSTRACT OF THE THESIS

CUSTOMIZED WIRELESS MESH ROUTING METRIC FOR SWARM OF DRONES APPLICATIONS

by

Oscar Bautista Chia

Florida International University, 2020

Miami, Florida

Professor Kemal Akkaya, Major Professor

With the proliferation of drones applications, there is an increasing need for handling their numerous challenges. One of such challenges arises when a swarm-of-drones is deployed to accomplish a specific task which requires coordination and communication. While this swarm-of-drones is essentially a special form of mobile ad hoc networks (MANETs) which has been studied for many years, there are still some unique requirements of drone applications that necessitates re-visiting MANET approaches. These challenges stem from 3-D environments the drones are deployed in, and their specific way of mobility which adds to the wireless link management challenges. In this thesis, we consider the existing 802.11s wireless mesh standard and adopt its routing capabilities for swarm-of-drones.

Specifically, we propose two link quality routing metrics called SrFTime and CRP metrics as an improvement to the 802.11s default Airtime routing metric, to enable better network throughput for drone applications. SrFTime improves network performance of stationary and mobile Wireless Mesh Networks, while CRP is designed to fit the link characteristics of drones and enable more efficient routes from these to their gateway. The evaluations in the actual 802.11s standard indicate that our proposed metrics outperforms the existing one consistently under various conditions.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

In recent years, Unmanned Aerial Vehicles (UAV) or drones have been used in many military and civilian applications such as search and rescue operations, detection and tracking, intelligent transportation systems, managing wildfire, relay deployment, and traffic monitoring [IBScT13][HYM16]. The trend in those applications indicates that typically a swarm of small-size UAVs are deployed due to its advantages for handling various tasks in coordination, when compared to a single large-size UAV in terms of cost, scalability, survivability, the speed of task completion, and small cross-section coverage [IBScT13]. In such applications drones carry sensors or can be in touch with other IoT devices in the environment for various tasks.

A swarm of drones is sometimes referred to as Flying Adhoc networks (FANET), which is synonymous to Mobile Ad-hoc Networks (MANET) that have been studied heavily in the past [MSS13][QBN+19]. However, there are a number of differences which distinguish FANETs as a subset of MANETs while they certainly share many similar characteristics. For instance, FANETs typically have a much higher mobility with unpredictable movements that may result in frequent topology changes, while "MANET nodes usually have very low mobility" [GC18]. In a FANET, UAVs may be equipped with multiple sensors to collect data from their surrounding and then relay it to a control center [SAU18][SAAU18] which is akin to wireless sensor networks (WSNs) [SAI+18]. Consequently, besides supporting peer-to-peer communications among drones for coordination and cooperation to maintain the network formation, FANETs also need to support data traffic that may require different data delivery strategies or quality of service (QoS) requirements. Furthermore, FANETs may also

need to operate in a rapidly changing environment in 3-D terrain settings, from close to the ground up to high in the sky. 3-D settings are interesting as they may influence the number of links and interference among different nodes. Thus, to support reliable and stable FANET operations in various settings including urban and rural environments, MANET standards/protocols may not be directly applied.

One of such example MANET cases is when drones need to communicate with each other (i.e., meshing among each other) when they cannot communicate with an existing ground infrastructure or when such infrastructure is not available [SAAU18]. In such cases, for supporting multi-hop meshing among drones, routing protocols are needed. As this has been a vast area of research for MANETs and some standards such as IEEE 802.11s [80211], Zigbee, etc. are already developed, they can be deployed for the same purpose in FANETs. Particularly, 802.11s standard suits the high data needs of drones as opposed to Zigbee or others. In addition, Wi-Fi dongles to run IEEE 802.11s are already available and they are convenient to use without any additional effort. Indeed, recently, this protocol was included in Google's WiFi routers as well [Goo20].

However, IEEE 802.11s was not designed specifically for FANETs. It was mainly geared for stationary WiFi nodes that can form a wireless mesh network (WMN) to access the infrastructure. In a sense, it gives nodes the capability to do multi-hopping if one-hop communication to an access point (AP) is not available. Nevertheless, the existing works on drone routing [SAU18][SAAU18][KGBV17] use the IEEE 802.11s standard as is which do not take into account the 3-D nature and abrupt link changes in such topologies. The way routes are determined is based on a certain routing metric called *Airtime* using an advanced on-demand distance vector (AODV)-like [PBRD03] routing protocol called Hybrid Wireless Mesh Protocol (HWMP) [80211]. Basically, *Airtime* metric is derived from the time resources used

by a specific link to send a test packet across the link and the average frame error rate on the same link.

In this research work, we argue that this current metric for HWMP routing does not fit well to FANETs due to their special challenges and requirements. We therefore propose a new routing metric that will best suite the needs of FANETs. The first improvement to the existing metric is called *Square Root Frame Time* (*SrFTime*) which is computed by using the existing *Probe* packets in IEEE 802.11s standard. Basically, we redefine the combination of time resources, error fluctuation and interference of links in 3-D environments to this new metric to improve certain QoS performance such as network throughput. Since the new metric does not require additional communication, no network overhead is added making this a cost-efficient routing metric too. Next, we built upon this metric to derive a *Comprehensive Radio and Power (CRP)* metric optimized for mobility. *CRP* minimizes some of the negative effects of link breakage due to mobility of drones that include increased packet bufferization and delay while a path repair is completed. CRP influences the metric value of links to nodes that are close to the transmission range coverage limit based on received power level measurements in an attempt to use alternative and more stable routes preferable over the current routes before a link break occurs.

We incorporated the proposed *SrFTime* and *CRP* into 802.11s mesh standard at the MAC layer in ns-3 network simulator. We then evaluated the performance of the revised IEEE 802.11s in 3-D mobile FANET topologies that we built based on a proposed group mobility model. This model was adapted from an existing reference point group mobility model to accommodate the characteristics of drones in a swarm. The results under a variety of conditions indicate that our proposed metrics consistently outperforms the original airtime metric in both stationary and mobile FANET topologies and turn out to be a viable option for practical deployment.

CHAPTER 2

RELATED WORK

There has been a lot of studies on routing and routing metrics for MANETs and WSNs to improve their performance. In this section, we summarize them to compare with what we are proposing.

2.1 Routing in Mobile Wireless Networks

Routing in MANETs and Vehicular Adhoc Networks (VANETs) have been widely studied [KRN10, LW07]. However, both of these types of wireless network have different routing requirements and challenges. They are not designed for 3-D environments, their mobility patterns and speeds are different and their data patterns are specific. Therefore, they need to be adapted for FANETs.

To this end, A. Nayyar [Nay18] performed a comprehensive comparison of routing protocols for MANET such as AODV [PBRD03], DSDV [PB94], DSR [JMB07], AOMDV [MD01], OLSR [CJ03] and HWMP [80211] when they are deployed for FANETs though the testing mobility model was not specified. The results for packet delivery ratio against node speed shows that HWMP outperformed the other routing protocols. Additionally, HWMP scored the highest throughput while sharing the least end-to-end delay performance with DSDV.

Due to the ability of HWMP to perform well for FANETs, we opted to improve its performance under a more realistic mobility model for 3-D FANET environments. In addition, it is already part of the IEEE 802.11s standard which has been implemented in practice.

2.2 General Routing Metrics

Routing metrics have been widely studied in many contexts for MANETs, WSNs and other similar networks. In this regard, G. Parissidis et al. [PKB+09] made a comprehensive survey and analysis of different routing metrics for WMN, categorizing them using different criteria such as the optimization goal including minimize delay, maximize network throughput among others. The authors also made distinctions in the way the information for metric computation is collected. In this study [PKB+09] the authors concluded that Expected Transmission Count (ETX) performs better than other metrics like Round Trip Time (RTT) and Per-hop Packet Pair (PktPair) since it is load-independent. Nonetheless, they also mention that ETX does not take into consideration the transmission rate in multi-rate ad hoc wireless networks. Expected Transmission Time (ETT) was developed by adding to ETX a factor that included the size of a probing packet divided by the bandwidth of the link. Medium Time Metric (MTM) was designed independently around the same time by B. Awerbuch et al. [AHR04], which is very similar to ETT differing only in that it includes control frames, back-off and fixed headers in the calculation of the link time usage, making MTM almost equivalent to the IEEE 802.11s' default Airtime metric. ETT and MTM throughput outperforms ETX especially in multi-rate ad hoc networks. In our work, since we consider 802.11s-based FANETs, we propose improvements to Airtime metric to fit it to the requirements of FANETs.

2.3 Improvements to IEEE 802.11s

As our work involves revision to the existing IEEE 802.11s Airtime metric, we also summarize the literature on the improvements of this standard. The closest study to ours in this sense is reported in [PZ18]. The authors propose a modification

to 802.11s airtime link metric, considering the current link load in addition to the default elements used to compute airtime. They basically propose new airtime metric values for different transmission rates and link load. However, there is no actual implementation or simulation that assesses the proposed improvement. Our purpose in this work is different as we seek to develop an improved metric that outperforms the current Airtime metric in terms of network throughput specifically for FANET applications where we consider 3-D environments and dynamic nature of drone links.

2.4 Mobility Models

MANET and VANET mobility models differ significantly from FANETs and this might cause a significant impact on the effectiveness of the proposed solutions since the evaluation results may differ considerably from real deployments. A. Bujari et al. [BCC+17] has noted the importance of using appropriate mobility models for FANET simulation. Similarly, G. Litvinov et al. [LLK18] confirmed the same finding and also added coverage area and node density as important factors to consider for proper network operation. Specifically, group mobility models are more appropriate for FANET applications. However, to the best of our knowledge, there is no specific group mobility model designed for FANETs.

One of the group mobility models from the literature that might be adjusted for FANET simulation is Reference Point Group Mobility (RPGM) [HGPC99]. RPGM defines group of nodes where each group has its own mobility pattern and in each group there is a reference point, so that nodes belonging to that group will move pseudo randomly around this reference point with a defined maximum distance to it. Therefore, in this work, we chose RPGM to be adopted for swarm movement.

CHAPTER 3

PRELIMINARIES

3.1 System Model

We assume a swarm-of-drones or FANETs deployed for a mission in an area of interest, which could be either urban or rural. The drones are equipped with 802.11s radio communication interfaces. One of the drones in the swarm will be acting as the root, which is assumed to have long distance communication capabilities such as LTE to communicate with a control center, serving as a gateway for the 802.11s network. The drones may need to exchange data as well as relay data of each other to the gateway if the data needs to be communicated to a remote location. Fig. 3.1 exemplifies such scenario. For the rest of the manuscript, swarm and FANET are used interchangeably. Next, we provide some preliminaries before we explain our approach.

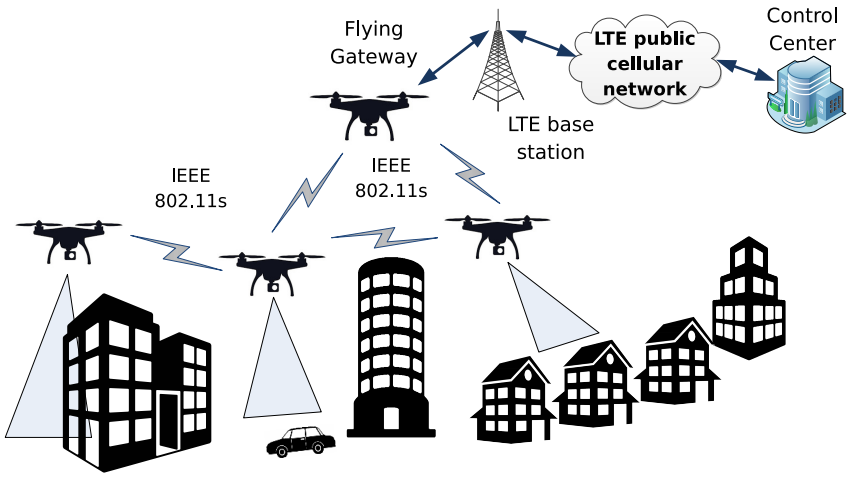


Figure 3.1: Drone Deployment in Urban Environment

3.2 Hybrid Wireless Mesh Protocol

Hybrid Wireless Mesh Protocol (HWMP) is the default mandatory Routing Protocol for the 802.11s standard, though it allows vendors to implement any routing protocol and path metric as well [AB11]. HWMP supports two modes of operation: Reactive and Proactive modes, that can be used concurrently. The Reactive (On Demand) mode is adapted from AODV Routing Protocol [PBRD03], which initiates a path discovery when a station has a data to transmit. HWMP Path selection frames are used for path management. The basic frames are: *Path Request (PREQ)*, *Path Reply (PREP)* and *Path Error (PERR)*. The first two are used for path discovery, while PERR is used to eliminate paths under specific conditions.

When a station needs to send data, it broadcasts a PREQ. Neighbors that are not the intended destination in turn will forward the PREQ to its neighbors propagating the PREQ throughout the network. The PREQ is updated at each station adding its new *link metric* value to compute a *path metric*. Note that the link metric is computed independently by each station. The stations receiving the PREQ that are not the intended destination but know a path to it could also send a PREP back to the station the PREQ was received from depending on the flags in the PREQ frame. PREP is a unicast frame. When the PREQ eventually reaches the intended destination, it sends a PREP back to the originator, following the same PREQ path. The best path (the one with the lower metric) is chosen at origin.

The proactive mode is used optionally. In this mode, one node in the network is chosen as the root node, which finds routes toward all nodes in the network by broadcasting proactive PREQs periodically. This way, when a node needs to send data to other stations, it first looks up the destination in its own routing table. If an entry does not exist for that destination, then it sends the data packet to the root

node which relays the packet to the final destination. A combination of reactive and proactive modes enables efficient path selection in a wide variety of mesh networks [AB11].

3.3 Airtime Link Metric

The 802.11s Wireless Mesh Standard uses *Airtime* as the default Routing Metric. According to the specification, the cost of a link that uses this metric has two main components: the time that would take a test frame to be sent through the link (B_t/r) and the average frame error rate (FER) of that link (e_{fr}) as shown in Equation 3.1:

$$C_a = \left(O + \frac{B_t}{r}\right) \times \frac{1}{1 - e_{fr}}, \tag{3.1}$$

where O is the Physical Layer (PHY) dependent channel overhead that consists of the frame headers, training sequences, and access protocol frames. O is defined as $O = O_{ca} + O_p$, where O_{ca} is the channel access overhead, and O_p is the protocol overhead. The parameter B_t is the size of the test frame in bits, whose default value is 8192 (or 1024 bytes), and r is the physical link rate. Table 3.3 shows the Airtime Metric Constants for some of the 802.11 standards [CK08][ABB10].

	IEEE 802.11a	IEEE 802.11b/g
O_{ca}	$75\mu s$	$335\mu s$
O_p	$110\mu s$	$364\mu s$

Table 3.1: Airtime Metric Constants

The 802.11s standard does not define a specific method to calculate the frame error rate, e_{fr}. Rather, it is left to the specific implementation.

The final result of the metric is expressed in units of 0.01 Time Unit (TU) = $10.24\mu s$, as required by the standard.

3.4 Propagation Loss Models

Propagation Loss Models are used to compute the receive Rx power level of the radio signal considering a specific environment. The two models we used for testing in this paper are briefly described below:

3.4.1 Friis Propagation Loss Model

The Friis model [Fri46] uses a particular case of the equation found in "A Note on a Simple Transmission Formula", expressing the effective area in terms of the gain as defined in Equation 3.2.

The Friis model is valid only for propagation in free space within the so-called far field region, which can be considered approximately as the region for $d > 3\lambda$.

$$\frac{P_r}{P_t} = G_t G_r \left(\frac{\lambda}{4\pi d}\right)^2, \tag{3.2}$$

where:

P_r : Power available at the output terminals of the receive antenna.

P_t : Power fed into the transmitting antenna at its input terminals.

G_t : Transmission Antenna Gain

G_r : Reception Antenna Gain.

λ : wavelength.

d : Distance between antennas.

In practice, however, Friis is often used in scenarios where accurate propagation modeling is not deemed important, and values of $d = 0$ can occur. To do that, the implementation of the Friis model provides an attribute called MinLoss which allows to specify the minimum total loss (in dB) returned by the model [ns-18].

3.4.2 ITU-R1411 Propagation Loss Model

This model implements the ITU-R recommendation for Line-of-Sight (LoS) and Non Line-of-Sight (NLoS) short range outdoor radio communication in the frequency range from 300 MHz to 100 GHz.

This recommendation is based on the premise that propagation over paths of length less than 1 km is affected primarily by buildings and trees, rather than by variations in ground elevation. The effect of buildings is predominant, since most short-path radio links are found in urban and suburban areas [Int17]. The standard consider cases for urban high-rise, urban low-rise, roof-top suburban, residential and rural. This is important for the simulation of FANETs. The propagation behavior of these models is symmetric in the sense that they treat radio terminals at both ends of a path in the same manner, from the model's perspective, it does not matter which terminal is the transmitter and which is the receiver.

In our simulations we implemented the ns-3 LoS model since it accurately reflects the path loss in real environments. This model exhibits the particularity that the link attenuation between two nodes separated by a fixed distance changes at different nodes altitude, this is, the higher the nodes the lesser the link attenuation.

CHAPTER 4

SRFTIME ROUTING METRIC

4.1 Motivation

In IEEE 802.11s standard, the default Routing protocol HWMP, finds possible paths between a source and a destination then selects the best of them based on the metric also known as path cost. The path cost is calculated by adding the cost of each link along the way from source to destination and the preferred path is the one with the lower cost. As described in the previous section, the *Airtime* link metric quantifies the link quality based on the time it takes for a test frame to be transmitted through that link and weighted by the frame error rate. One link is preferred over another when its metric value (cost) is lower. Typically, a lower *Airtime* metric value is obtained when link speed is higher and frame error rate is lower.

However, airtime metric may not always be optimal for FANET applications. For instance, after close analysis of the effect of link rate in the airtime metric, we found that the discrete changes in the airtime link metric value due to a link rate change combined with specific average frame error rates may not result in a better throughput which is becoming more crucial for drone applications where video data collection, sensing, command&control are becoming very common.

Therefore, in this work, we first introduce a revised routing metric called *SrF-Time*, geared for increasing the FANET performance in terms of network throughput. For this purpose, we propose alternative ways to compute the average FER and the link time usage as explained next. By using this metric, we expect that the network performs more efficiently by reducing collisions altogether.

4.2 Modified Frame Error Rate Calculation

The 802.11s standard does not define a specific method to calculate the average frame error rate. As a specific implementation example, ns-3 simulator uses an exponentially weighted moving average, where the last known average frame error rate is weighted by a coefficient that decreases exponentially with time, and the result of the most recent transmission have more weight. One disadvantage of this method is that it depends on user data transmissions, and thus the measure is not accurate during idle periods.

In this paper, we propose an alternative method to measure the frame error rate on a link. Specifically, this involves sending *probe* packets between neighbors, and measuring how many of those packets have been received in the forward and reverse direction for each link during a given period. Nonetheless, there would be a drawback of increasing the network overhead when implementing this method which may be counter-productive in improving throughput.

Therefore, we need to come up with a method that does not add any overhead by avoiding the transmission of additional frames. To this end, we propose using information already sent by every node in the network through the use of *Beacon* frames, which are already part of IEEE 802.11s mechanisms. Specifically, this is possible via a protocol called *Peering Management Protocol (PMP)* [AB11].

Apart from HWMP, in 802.11s by default for path discovery and maintenance, PMP is other important protocol, which is responsible for neighbor mesh station (peer) discovery and link management. The PMP opens new links to neighbor stations and also closes links when failures on them are detected. Mesh stations are not allowed to transmit frames other than the ones used for peering management (Open/Confirm/Close) to a neighboring mesh station until a corresponding link

has been established [AB11]. To make neighbors aware of its presence, every mesh station sends periodically small one-hop management frames, known as *beacons*. Each beacon broadcasts information about the station capabilities which includes supported rates, extended supported rates in addition to what is important for our purpose: the beacon timing containing the beacon interval and the time at which the reporting station last received a beacon from every neighbor. In this way, we will be able to compute average frame error rates in forward and reverse directions.

Fig. 4.1 shows an example for this calculation with a network of 6 nodes and considering the link between nodes A and B specifically. Both A and B broadcasts beacon frames at a specific interval. By defining a window size and counting the number of beacons received during that time window, node A can get a frame delivery ratio from B to A. For instance, if the window size is set to receive 10 beacon intervals in the ideal case and 1 beacon is missed for whatever reason, then the frame delivery ratio in the forward direction (B to A) is $f_{df} = 9/10$. The same equation applies to calculate the frame delivery ratio in the reverse direction f_{dr}: Node A keeps a record of timestamps read from the beacons so that it can obtain a count of its beacons received by node B.

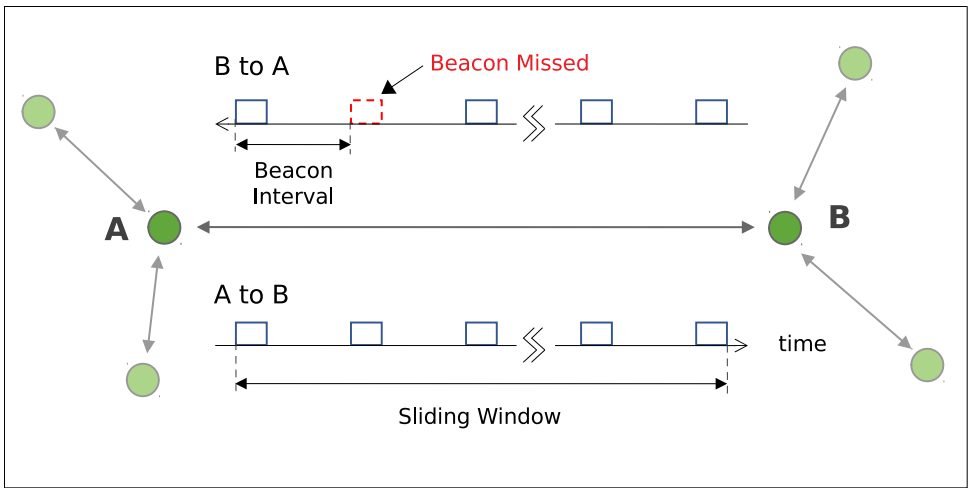


Figure 4.1: Proposed Frame Error Rate computation from Beacons.

Consequently, putting these together, we can compute an extended bidirectional average frame error rate, ex_{fr}, as defined in Equation 4.1 to be included in $SrFTime$, where f_{df} and f_{dr} are the frame delivery ratio in the forward and reverse direction respectively:

$$ex_{fr} = 1 - (f_{df} \times f_{dr}) \tag{4.1}$$

This is inspired from the approach of ETX computation. Basically, we multiply f_{df} and f_{dr} to compute average delivery ratio in both directions of the link and then we subtract it from 1 to come up with the error rate.

Details of the implementation of this new method to derive an average FER is given in Chapter 6.

4.3 Link Time Usage Calculation

The second modification to the default airtime link metric calculation is related to the airtime portion specifically. We would like to motivate this modification with a concrete example. According to Equation 3.1, the default airtime metric for a 1Mbps link with no errors is 887, and 470 for 2Mbps. Next, let us assume that the 2Mbps has a average frame error rate of 46%, the resulting metric is now $470/(1 - 0.46) = 870$. If one mesh stations has two possible paths to follow, it could still prioritize the 2Mbps link over the 1Mbps link ($870 < 887$) even when the first one has a frame error rate of near 50%. A similar case occurs if we compare a 12Mbps link with 39% frame error rate $107/(1 - 0.39) = 175$ to a 6 Mbps with no errors whose default airtime Metric is 176.

Our approach for the new metric allows more contribution from the error rate component by eliminating the direct proportionality between time in the air (B_t/r) and the metric value in Equation 4.2, where α and β are weighting factors.

$$SrFTime = \left(\alpha O + \beta\sqrt{\frac{B_t}{r}}\right) \times \frac{1}{1 - ex_{fr}} \tag{4.2}$$

Basically, we propose to get square root of B_t/r to reduce its impact on the overall metric. Moreover, we tuned the values of α and β in our implementation. After comparing the results of several experiments, it was determined that $\alpha = 1$ and $\beta = 20$ provide a sustained improved performance. We maintain the requirement to have the final metric divided by 0.01 TU or $10.24\mu s$ as per the standard. The new metric values for different link rates and zero frame error rate are shown in Table 4.3. As seen, we were able to reduce the values of Airtime metric, which will eventually impact the routing decisions for improved throughput.

Modulation	Default Airtime	Proposed SrFTime
Dsss1Mbps	887	217
Dsss2Mbps	470	165
Dsss5_5Mbps	205	117
Dsss11Mbps	130	96
ErpOfdm6Mbps	177	110
ErpOfdm9Mbps	131	96
ErpOfdm12Mbps	108	88
ErpOfdm18Mbps	84	79
ErpOfdm24Mbps	73	73
ErpOfdm36Mbps	61	67
ErpOfdm48Mbps	55	63
ErpOfdm54Mbps	53	62

Table 4.1: Airtime Metric Values for 802.11g

The original *CalculateMetric()* method in the ns-3 model, also contains the modification for the calculation of the *SrfTime* and *CRP* metrics according to Equation

4.2. The B_t/r portion is returned by the method $mac \rightarrow GetWifiPhy() \rightarrow CalculateTxDuration()$. More details of it in Chapter 6.

CHAPTER 5

CRP: A ROUTING METRIC OPTIMIZED FOR MOBILITY

SrFTime metric aimed at improving network throughput when the nodes are stationary. However, as mentioned earlier FANETs may exhibit high mobility depending on the applications, which causes an increase in the number of link changes. For instance, some links may be broken eventually due to movements while new links are established. Therefore, in order to improve the network performance, it is imperative to account for variations in the links caused by the change of relative positions between nodes and incorporate those indicators into the link metric.

A potential intuitive approach to account for mobility effects in the metric is to use the relative location and velocity of neighboring nodes through GPS. Since drones have built-in GPS used for location and flight control, a drone's location and velocity could be shared with neighboring drones to calculate how far they are from each other and estimate their relative position after a brief period of time. Nonetheless, physical location by itself cannot be used to determine with good precision when a link is about to be broken because a given transmission (Tx) power and link distance will produce a different reception (Rx) Power at the receiver node depending on many factors. These factors include but not limited to the type of environment (rural, urban, etc.), and the nature of obstacles and weather/atmospheric conditions blocking Line-of-Sight (LoS) capabilities. Therefore, we opt to rely on Rx power as a more reliable indicator to accommodate mobility effects.

Specifically, when two nodes get far from each other, the attenuation of the radio signal that travels between them increases, resulting in less Rx Power. The consequences of a reduced Rx Power include reduced signal to noise ratio (SNR) and link breakage if Rx Power drops below the receiver sensitivity level also known as Energy Detection Threshold (Ed_T). In our case, because of the high mobility of

drones, it is possible that two nodes that are communicating just fine could suddenly lose connection because of loss of signal (i.e., Rx power drops below Ed_T). While mobility may not be the sole reason for this, it is the main cause and it needs to be accounted for both sides considering that in a mobile mesh network all nodes are typically configured with the same transmission power.

Based on these observations, our revised metric utilizes the Receive Signal Strength Indicator (RSSI) which is the power level of the received frame as a more accurate method to determine when a node is nearing the transmission coverage limit. Therefore, we modify the SrFTime link metric further by adding a new component that will increase the metric value when the RSSI value is so close to Ed_T as an indicator that a link break is very likely to occur. Furthermore, we note that we do not increase the link metric when RSSI value is within normal values. In other words, when the nodes are within good range from each other as determined by RSSI, we let the best route to be chosen by SrFTime metric, and only when the nodes are close to the transmission coverage boundaries inside a 'warning zone', we would like to increase the link metric so that the nodes considering this transit link eventually select a more suitable one assuming there are alternatives before a link break occurs. This is depicted in Fig. 5.1 where we show a reference node as a blue circle and its four neighboring nodes. The orange neighbor is located in the warning zone which we define as the zone within k dB from the coverage limit where k could be set experimentally. For nodes in that region, we want to increase the value of the link metric (i.e., the link will not be picked by the routing algorithm) as it is prone to be out of range at any moment.

Considering that in a mobile mesh network all nodes are configured with the same transmission power, when a received packet's power level is close to the Ed_T we can assume that the the counter part node is also near the limit of our coverage

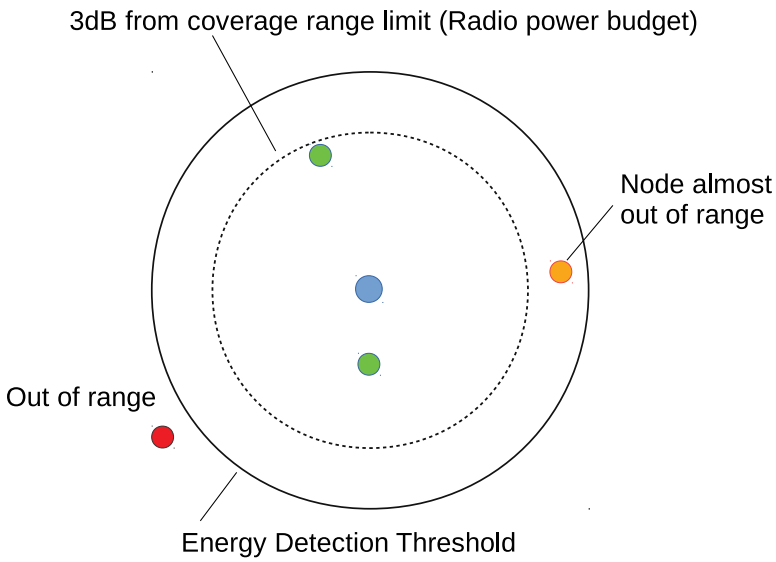


Figure 5.1: Node Coverage Range and Warning Zone

region.

Putting it all together, we can add this enhancement due to mobility to our SrFTime metric which is called Comprehensive Radio and Power (CRP) metric hereafter. CRP basically aims to take into consideration the likeliness of a link break when there is a small margin between the Rx power and the Energy Detection Threshold Ed_T as stated in Equation 5.1:

$$CRP = \begin{cases} SrFT + \gamma \left(10^{\frac{k-PB}{10}} - 1\right) \times \frac{1}{1-ex_{fr}} & \text{if } PB < k \\ SrFT & \text{otherwise,} \end{cases} \tag{5.1}$$

where γ is a weighting constant and PB is the power budget in dB defined by:

$$PB = RSSI - Ed_T \tag{5.2}$$

Note that Ed_T is a value configured in the network device and it is available to use. The determination of the optimum k threshold is based on experiments.

By evaluating different values and picking the threshold that resulted in better performance we came up with a k threshold of 3dB.

CHAPTER 6

NS-3 IMPLEMENTATION DETAIL

6.1 Frame Error Rate Calculation

We implemented an average FER measurement using information in the *Beacon* frames of 802.11s, which are sent periodically by every mesh station to make their neighbors aware of their presence. By following this approach, we avoid increasing the network overhead and obtain a FER that improves overall network performance.

Note that a beacon frame contains information about the node capabilities in order to establish a link. Apart from information of supported rates and extended supported rates, the beacon also contains other information useful for our purpose such as: the beacon time containing the beacon interval (period) and the time at which the station received the last beacon from each of its neighbors. By counting the beacons received within a specific interval and processing the information contained in it, we can compute a bidirectional average FER as shown in Eq. 4.1 to use in the default or additional link metrics.

To implement this metric in ns-3, a new `UpdateBeaconReceived()` method was created in the `PeerLink` class. To explain the use of this class, let us introduce another 802.11s protocol that runs in parallel to HWMP called Peer Management Protocol (PMP). PMP establishes or breaks links between nodes and its operation is supported by the beacon frames. When two nodes receive beacons from each other, the PMP starts the process to establish a new link. Furthermore, if a defined number of consecutive beacons are not received from a peer node with an existing link, then that link is broken by PMP. Routing or data frames are only transmitted between nodes when a link between them is established.

The `PeerLink` class is used by the PMP to maintain information and statistics for each link. Therefore, it is deemed appropriate to implement FER in this class. The new method is called from existing method `PeerManagementProtocol:: ReceiveBeacon()` that processes beacons received.

The time window size can be configured via set-attribute, with a maximum size of 30. The Average FER for each link is stored in `MeshWifiInterfaceMac` where a new structure was created for that purpose. That structure is accessed by the `AirtimeLinkMetricCalculator:: CalculateMetric()` method when a new link metric calculation is needed as shown in Fig. 6.1.

For each neighbor link, a timer `m_beaconMissedTimer` is set according to the beacon interval. The timer is reset when a new beacon arrives. If the timer expires, it means a beacon was lost for this link, and the count is updated accordingly. There is an anti-collision mechanism that shifts the next beacon transmission negatively or positively by a small time value. This happens when a beacon is not received as expected according to the beacon interval. In such case, it is assumed that the beacon has collided, therefore the next beacon transmission occurs at a slightly shifted time. To maintain accuracy, the new implemented `m_beaconMissedTimer` also considers that maximum beacon shift time value.

When one or more beacons are not received, the information about beacons received in the opposite direction is not available. In that case, the count is updated when the next beacon is successfully received.

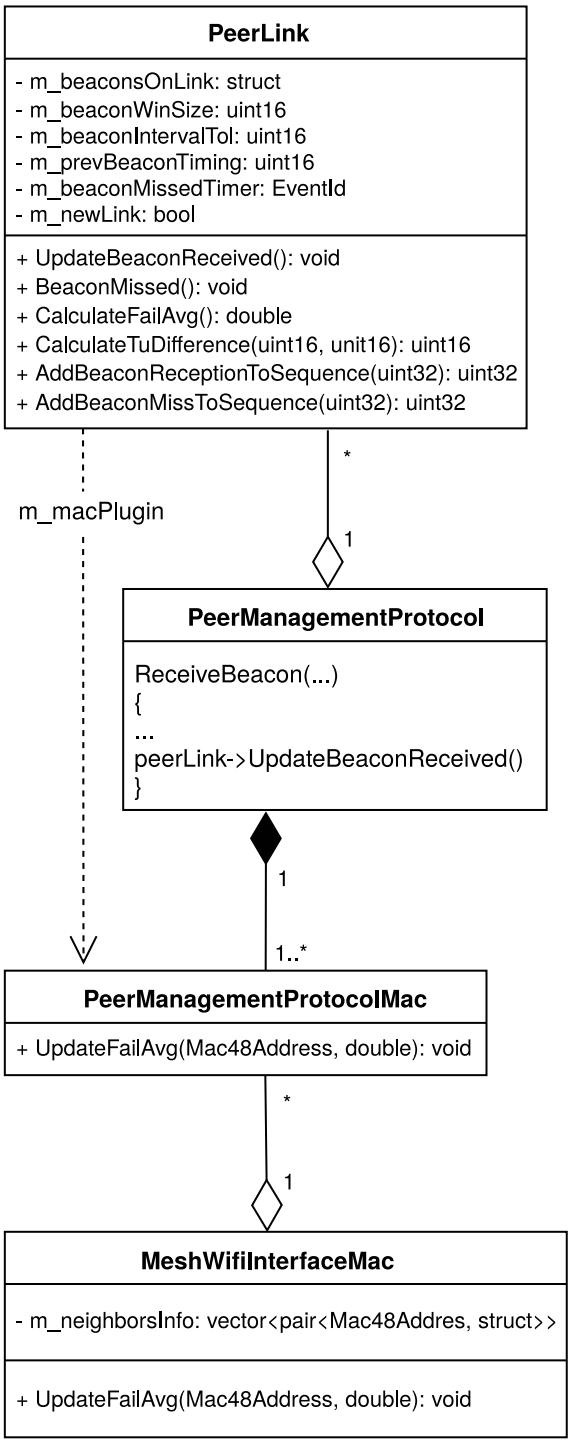


Figure 6.1: Methods and Variables added for Calculation of FER from Beacons

6.2 Implementation of New Link Metrics

In 802.11s, the default Routing protocol HWMP finds paths to a destination by adding the metric value of each link along the path. Finally, the source node selects the best path based on the path cost, which is the sum of all link costs. The path with the lowest cost is the one added to the routing table.

In ns-3's mesh model, the `MeshWifiInterfaceMac` class creates a callback to the `AirtimeLinkMetricCalculator:: CalculateMetric()` method, which in turn gets all the information required to compute the *airtime* link metric from the MAC interface the method was called from as shown in Fig. 6.2.

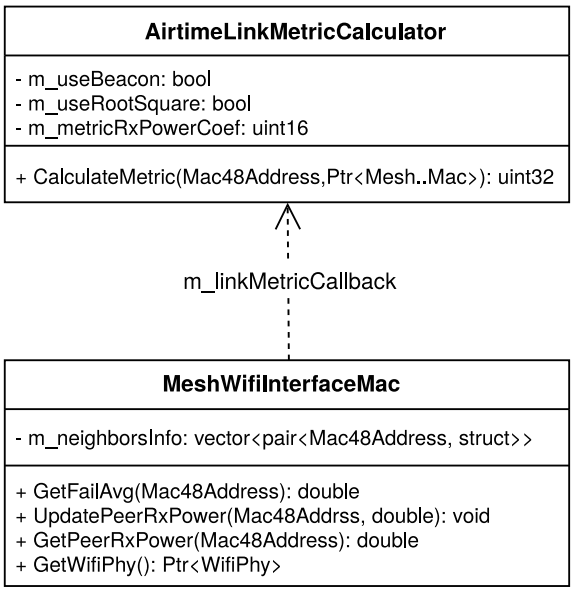


Figure 6.2: Class Diagram for the Computation of Link Metric

Having the link calculation implemented as a separate class facilitates the addition of other link metric options, provided that all the information required is accessible by that class. We implemented two proposed link metrics, namely *SrFTime* and *CRP* defined by equations 4.2 and 5.1 respectively. *SrFTime* provides

better performance in mesh wireless networks in general while *CRP* metric is optimized for networks with high mobility.

SrFTime uses the same input as the default *Airtime* link metric, except for the FER, which is calculated as described above and obtained from the MAC interface. On the other hand, *CRP Metric* requires additional information, which is the received frames' power level to affect the link metric calculation.

We followed a *cross-layer* approach in our implementation through tagging. The Remote Signal Strength Indicator (RSSI) is available from `YansWifiChannel::Receive()`, and added to a `ns3::Tag` class which is normally used to tag a packet or bytes of it with information without affecting the packet size. This way, the Rx power can be used in upper layers. When the packet is received at the MAC Layer `MeshWifiInterfaceMac`, the tag is removed and the Rx power value is stored for later use when a link metric is calculated. The Energy Detection Threshold Ed_T, is a value configured that can be retrieved from the physical layer `WifiPhy` through the MAC Layer.

CHAPTER 7

GROUP MOBILITY MODEL FOR FANETS

7.1 3D Mobility Scenario Generation

While it is important to customize a routing metric for FANET applications, it is also crucial to come up with realistic mobility models for FANETs that can be utilized in simulation-based testings since often actual deployment and testing of FANET research is very difficult. This need is also evident in the current research since most of the published research work on FANET performance experimentation considers trivial mobility models [BCC+17] that fail to represent a realistic formation and movement pattern of a swarm of drones. This is because traditional mobility models are purely random-based and does not exhibit such characteristics. Currently, there is no mobility scenario that has a strong correlation with the real mobility of a swarm of drones in 3-D environments. In addition, the work in [LGZ+19] reports that "the flight characteristics of UAVs in 3-D environment have been neglected, leading to inaccurate experimental results".

Consequently, the simulation of FANET node movements requires to use an appropriate mobility model that better represents the pattern and cinematic characteristics of drones, thus allowing to obtain more accurate results compared to other existing mobility models. In particular, since swarm of drones are used because of the many advantages they provide when performing cooperative tasks, their mobility should exhibit smooth changes in speed and direction, as well as group synchronization and [BCC+17] mechanisms for collision avoidance.

We claim that the best fitting mobility models for swarm of drones would be group mobility models [ZXG]. Although there have been some group mobility scenarios proposed to be used for drones, they still need some adjustments to mimic

the movements of an efficient 3-D mobile network. This is mostly due to the fact that the movement patterns in a swarm would follow a mixture of individual and group-based needs. In addition to these issues, there are difficulties with the implementation and availability of the proposed group models whether it is geared for drones or other applications. In most of the cases the corresponding scenario generator was - to the best of our knowledge - not published or available to use.

To this end, we picked the RPGM model for adaptation to FANETs. As introduced in Chapter 3, RPGM allows us to define groups which can have their own mobility patterns related to a reference point defined in each group. In addition, RPGM models could be generated by a mobility generator tool called BonnMotion [AEGPS10, oB16] which is available to use. However, RPGM had two problems when it comes to FANET adaptation. First, BonnMotion's RPGM was designed for 2-D MANET simulation rather than for 3-D scenarios. Second, in RPGM large size groups exhibit non-uniform distribution of nodes as depicted in Fig. 7.1. It can be observed clearly that the center of the area have a higher density of nodes and some of these nodes even get dangerously close to other nodes. In contrast, the nodes located far from the group center have a more sparse distribution, potentially isolated and disconnected from the swarm.

To adapt this model for FANET environments, a post-processing of the scenarios generated was required as follows:

- To create subgroups with a manageable number of nodes, so that dense distribution of several nodes is minimized and potentially isolated nodes are eliminated with the goal of obtaining more uniform node distributions.
- For each subgroup, linearly scaling each node's Z-coordinates to a suitable range for swarm-of-drones.

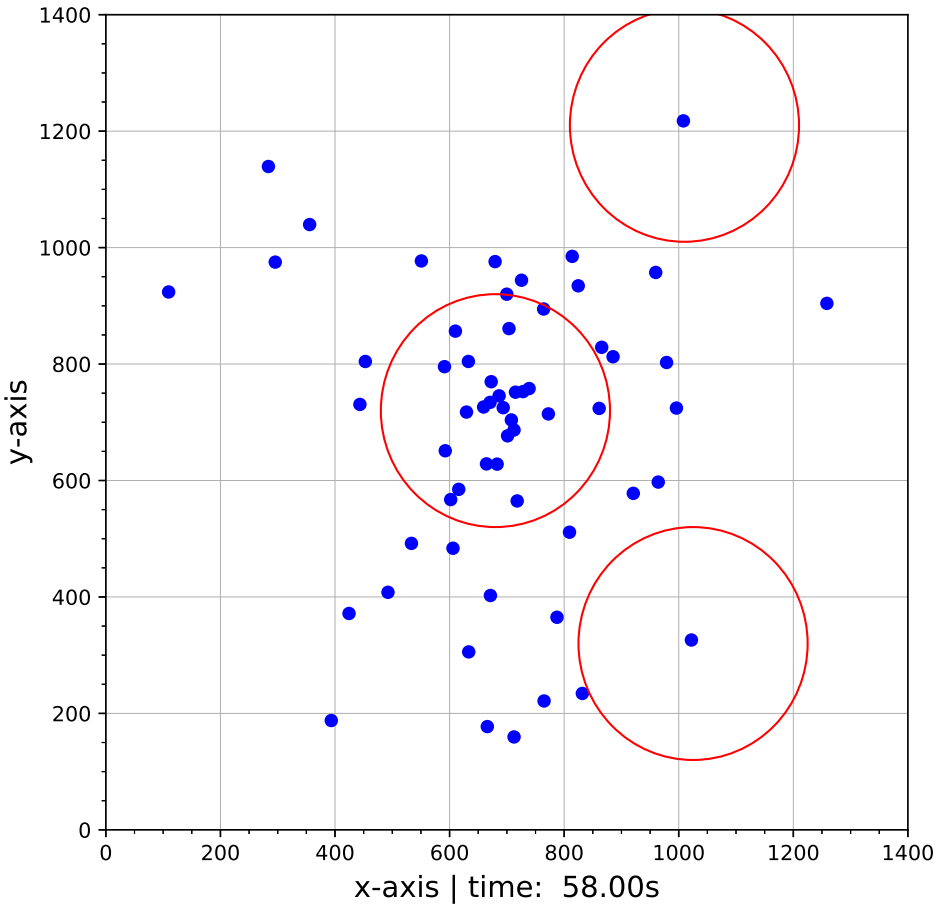


Figure 7.1: RPGM scenario with 60 nodes using BonnMotion

- Move each (x_i, y_i) within a subgroup i to a new location $(x_i + r_i, y_i + s_i)$ so that when placed in a common 3-D system they become adjacent groups with appropriate separation among each other. Here, r_i, s_i are the displacement values based on the actual distribution of nodes in subgroup i relative to the nodes in the common 3-D system.
- To merge all different subgroups into a single swarm. This is done by bringing all subgroups movement information into a single file in BonnMation's format.

Fig. 7.2 illustrates this process using an example of a 20-node mobility scenario created from 4 RPGM-subgroups with 5 nodes each. White circles are the reference

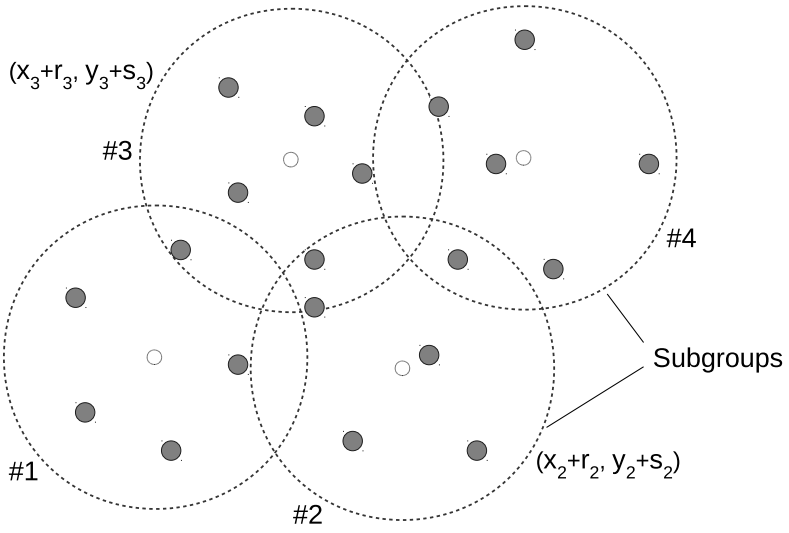


Figure 7.2: Group Mobility Scenario by merging RPGM subgroups

points for the subgroups. Note that when RPGM-subgroups are created, some nodes may be discarded to avoid extreme low or high density of nodes. Therefore, at the time of creation, the number of nodes per subgroup is chosen slightly higher than the desired number of nodes so that we can discard the ones that tend to deviate more from a cooperative group formation.

To compare with the topology in Fig. 7.1, we generated a new topology using our proposed mobility model as shown in Fig. 7.3 where we combined 6 RPGM-subgroups with 10 nodes each for a total of 60 nodes. We can observe how following the approach of sub-group merging ensures a better distribution of nodes. There can still be nodes that at some point in the simulation move close to other nodes, but the agglomeration is a lot less. Hence, it allows a more cooperative efficient network overall. Fig. 7.4 shows the 3-D view of the same topology with 60 nodes.

As part of this work, a tool with visualization capabilities was also developed to convert 3D mobility scenarios from BonnMotion format to NS2 mobility trace files format [Bau19b].

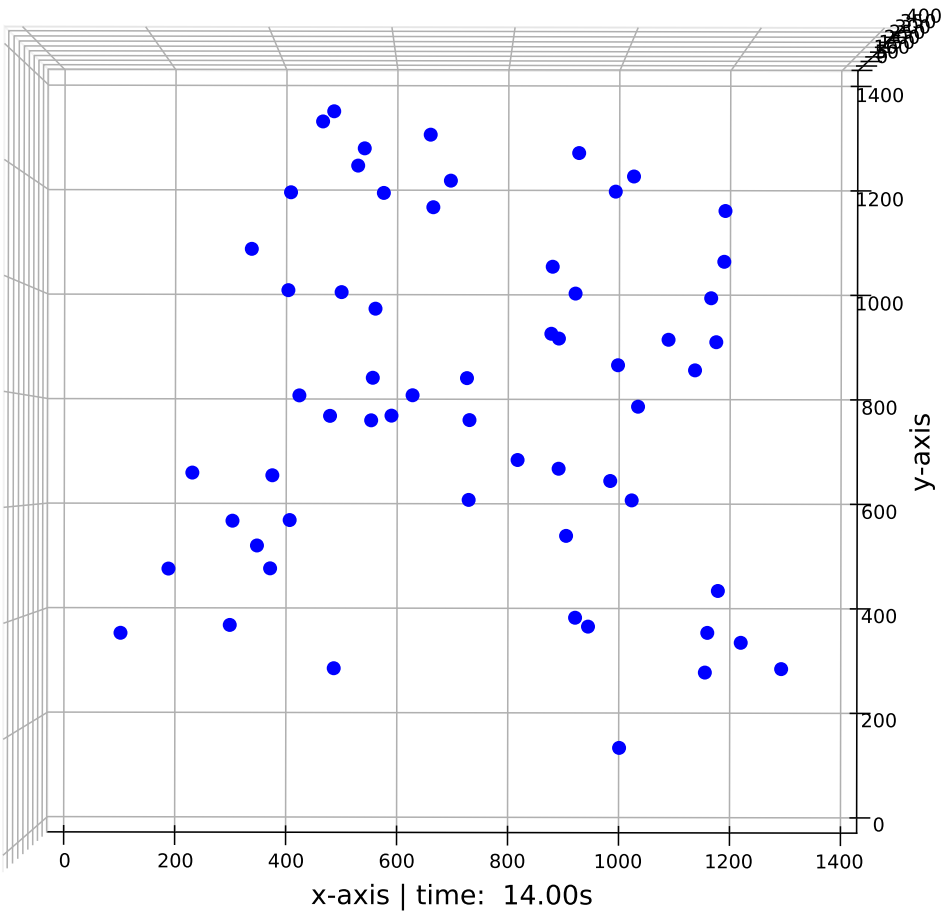


Figure 7.3: Upper View of 3-D Proposed Group Mobility Scenario

7.2 Mobility Model Implementation

For implementing the proposed mobility model adapted from RPGM, we used *Ns2MobilityHelper* class in ns-3 as a helper to import ns-2 (i.e., former version of ns-3) movement trace files for simulation into ns-3. Its use is very convenient as an alternative to the mobility models built in ns-3. Third party applications that generate mobility scenarios could export the location and mobility information in a compatible format that can be ported to other network simulators.

However, ns-2 movements were conceived for 2-D networks. This limitation is a hurdle for effective simulation in 3-D FANET that we needed to overcome.

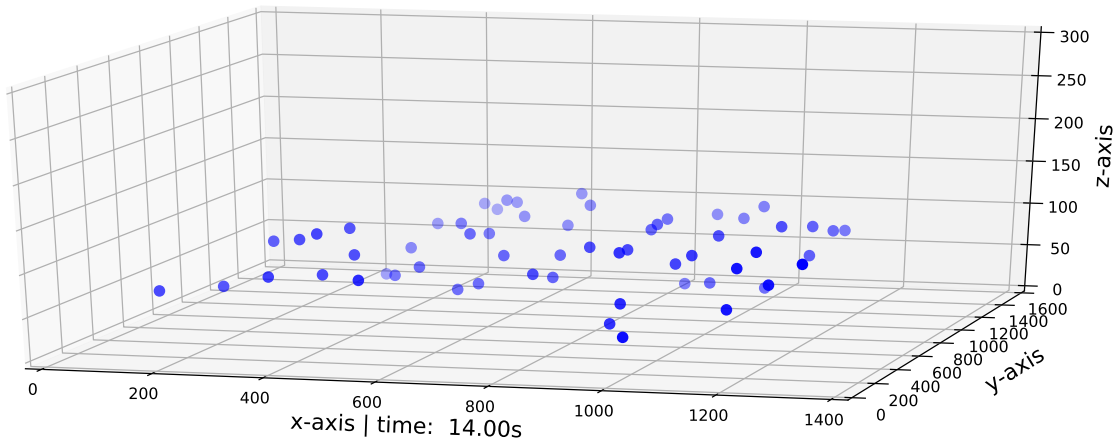


Figure 7.4: Example Scenario using Proposed 3-D Mobility Model

Therefore, the Ns2MobilityHelper files were upgraded to allow scenarios with full 3-D mobility.

These are the valid statements in the existing implementation of Ns2MobilityHelper:

```
$node set X_ x1

$node set Y_ y1

$node set Z_ z1

$ns at $time $node setdest x2 y2 speed

$ns at $time $node set X_ x1

$ns at $time $node set Y_ Y1

$ns at $time $node set Z_ Z1
```

The helper files were upgraded by adding to Ns2MobilityHelper class the capability to parse the following statement:

```
$ns at $time $node setdest x2 y2 z2 speed
```

With this improvement, suitable mobility scenarios for FANET simulation could be imported for simulation into ns-3 , and the updated files are freely available to use [Bau19a].

CHAPTER 8

PERFORMANCE EVALUATION

8.1 Simulation Setup

To determine and quantify the performance improvement of the proposed metrics, we implemented them within IEEE 802.11s module in ns-3 v3.29 [ns-18] network simulator. We followed a cross-layer approach in our implementation to retrieve information about Energy Detection Threshold and about RSSI through tagging.

We created random FANET topologies with 60 nodes to be used in the simulation. In these topologies, each station sends data at a constant bit rate to the root node, which is selected among the drones with the highest altitude in the swarm considering that this will be a gateway to another network in a complete real scenario. All parameters configured in ns-3 are shown in Table 8.1.

Parameter	Set to
Network Simulator	NS-3 (v3.29)
RemoteStaManager	MinstrelHt
Wifi Standard	802.11n 2.4 GHz
Max. Spatial Streams	1
Radio Frequency	2.437 GHz
E. Detection Threshold	-87 dBm
Traffic Pattern	Constant Bit Rate
Packet Size	536 bytes
Number of Nodes	60
Z Coordinate Range	30-120m
Propagation Loss Model	Friis, ITU-R1411
Stationary Data Traffic Time	100s
Stationary Simulation Area	2400x1200m^2
Mobile Data Traffic Time	120s
Mobility Model	Proposed Custom RPGM

Table 8.1: ns-3 Simulation Network and Test Parameters

We considered experiments for stationary (i.e., hoovering drones) and mobile drones separately and thus their setup was also different as explained below:

8.1.1 Stationary Nodes

The station's locations were selected randomly, representing a group of drones located inside an imaginary cube with dimensions 2400m x 1200m x 120m. All stations are configured with the same Tx Power Level depending on the propagation loss model (0dbm for Friis and -4dBm for ITU-R1411). Therefore, the topologies were adjusted so that there is a minimum distance between any two stations and also no station can be separated from the group for more than a maximum defined distance. Two sets of scenarios with different distance constraints were generated, one set with 100m/250m min/max and another set with 120m/230m. Each set with 30 different topologies add up to a total of 60 topologies or scenarios. Also, the minimum height for any station was set to 30m (the Z coordinate range selection makes a difference when using ITU-R1411 as the propagation loss model). The results provided correspond to the average of running the simulation tests over all 60 topologies to achieve statistical significance.

8.1.2 Mobile Nodes

In case of mobile topologies, we used five 3-D scenarios with a Z-coordinate range of 30m to 120m. This range is picked since some drones capture image or video from the terrain at lower heights while the upper nodes can be used to relay the communication. For generating drone locations randomly, we had some challenges. This is because the minimum and maximum distance between nodes cannot be easily controlled in RPGM and if the topology creation is not controlled, we may

end up with topologies that either do not have any routes to the gateway node or experience too much collision due to proximity of the nodes [LLK18]. Therefore, each topology was tested using at least 3 different Tx power values and we chose for each topology the results that produced the highest throughput collectively (i.e. for all routing metrics). By using the optimal Tx power, we ensure that in general nodes will have enough links to choose from when finding a path to the destination and thus allowing the conditions to better evaluate the performance of the routing metrics. The simulations tests are performed as in the case of stationary nodes where 59 mobile nodes are sending data to the root node. The results presented are the average of the results from all mobile scenarios.

8.2 Metrics and Baselines

The main metrics we used for assessing the performance are listed below:

- *Network throughput* which is determined by calculating the total data received by the root (gateway) node and dividing it by the time the nodes send the data.
- *End-to-end delay* which is computed by averaging all the end-to-end delay of the packets sent from every node in the network to the gateway node.
- *Route Changes* which indicates the number of route changes for all the nodes as an indicator of overhead. A route change is considered a change in the route table entry for packets destined to the root node.

For comparison, we used the default *Airtime* metric as the baseline. To validate the results under different scenarios and network variables, simulations were run considering both TCP and UDP as well as two propagation loss models: Friis and

ITU-R1411 LOS. Basically, Friis is just a baseline but ITU-R1411 LOS realistically reflects the path loss in FANET environments and its results will be crucial. As mentioned, we tuned the initial TX power for the nodes for each of these models to obtain an adequately connected network what will provide routes from each node to the gateway node.

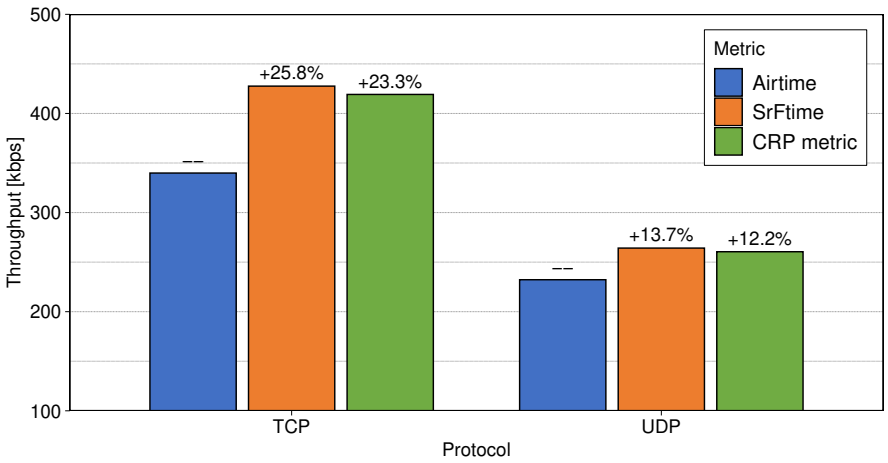
8.3 Simulation Results

We conducted experiments for both stationary and mobile topologies to see the impact. In this section, we first report results of stationary topologies and then move to mobile topology results.

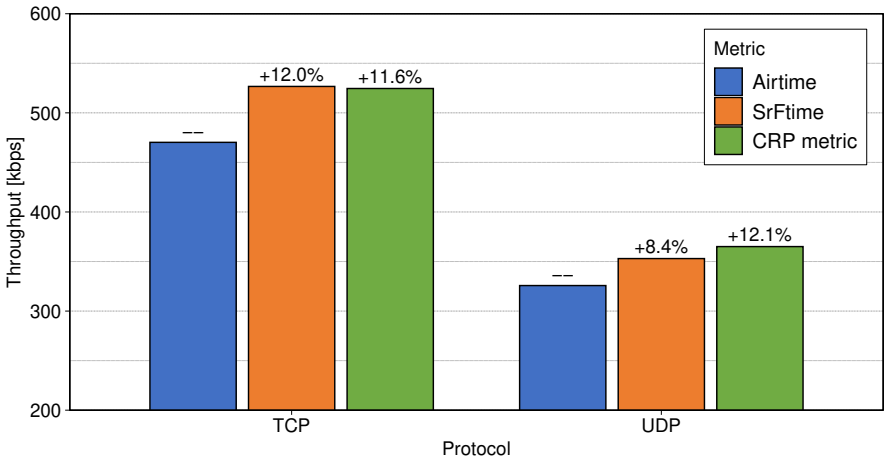
8.3.1 Throughput Results - Stationary

We first conducted experiments to see the impact of the proposed metrics on throughput. In these experiments, we varied the data rate (5/10/20/40kbps) from each drone for UDP and TCP traffic. We also collected results under different propagation loss models. The results are shown in Fig. 8.1.

We observed that under the Friis model, the network throughput increases about 26% in average consistently when *SrFtime* metric is used in routing of TCP traffic and about 14% for UDP traffic. The improvements follow a similar pattern for *CRP*, slightly behind the *SRFTime* metric. Similarly, under the ITU-R1411 model, we observe that our metrics still performs much better (i.e., about 12% for TCP and 10% for UDP) though it is not as much as in the Friis model. These significant improvements can be explained by the route changes with the proposed metrics. We observed that the number of route changes in the network is reduced about 17% in average when *SrFTime* and *CRP* are used as compared to *Airtime* as shown



(a) Friis Propagation Loss Model

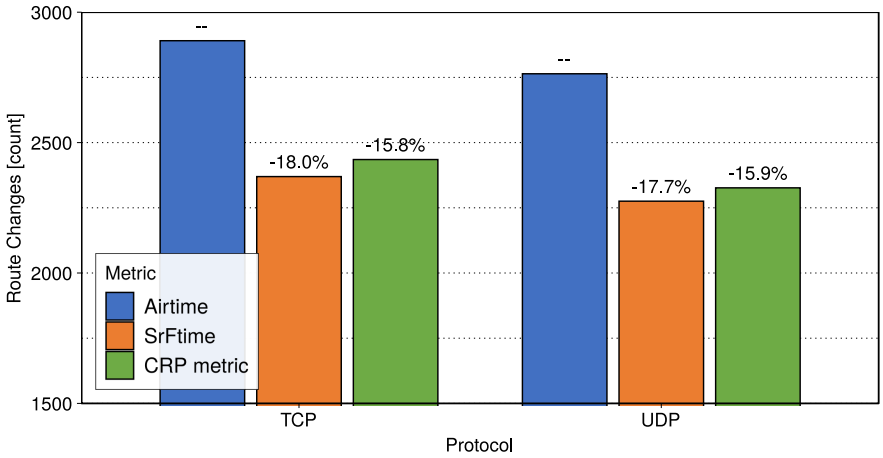


(b) ITU-R1411 Propagation Loss Model

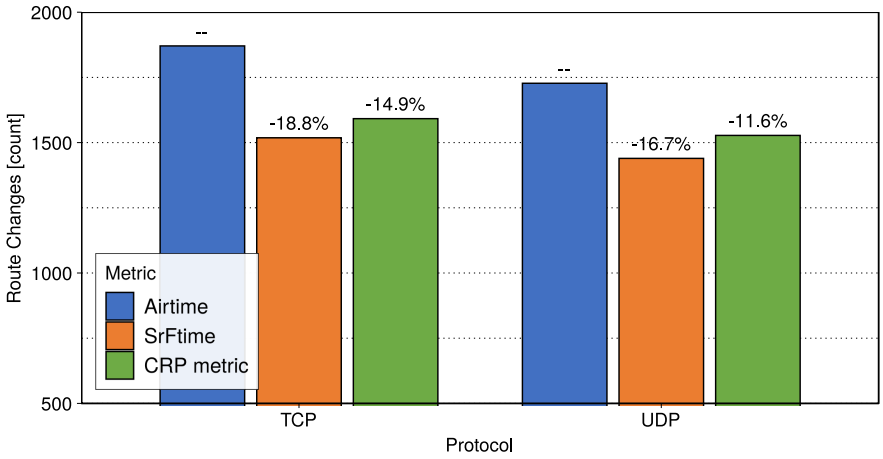
Figure 8.1: Network Throughput - Stationary FANET

in Fig 8.2. Specifically, it is known that some metrics suffer from self-interference [DPZ04], causing a negative effect in network performance. For the proposed metrics, the routes are therefore more stable which work in favor of increasing the network throughput.

When comparing *CRP* to the *SrFTime*, we observe that the latter is slightly better. This indicates that in general for stationary networks the longest links are important to have an efficient wireless network. By penalizing the longest links



(a) Friis Propagation Loss Model



(b) ITU-R1411 Propagation Loss Model

Figure 8.2: Number of Route Changes - Stationary FANET

with an additional component and forcing the routing protocol to avoid them when possible, we are decreasing the throughout slightly in the *CRP* case.

The other interesting observation is that the percentage in average network throughput improvement is higher for TCP compared to that of UDP. This might be attributed to the ways these protocols are designed. In case of TCP, when there is a route failure, there needs to be a re-transmission to enable reliability. In case of our metric, paths with higher reliability are chosen, which results in less re-

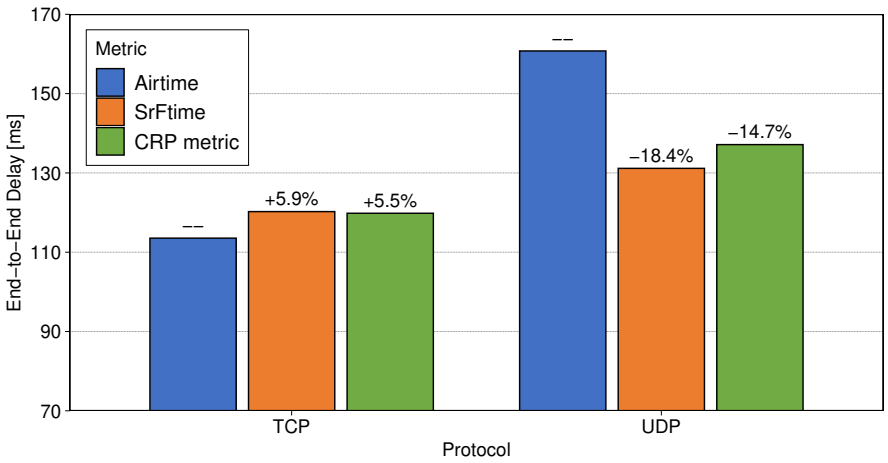
transmission. However, this is not the case for airtime metric where path reliability might be less. Re-transmissions cause a lot of overhead and may interfere with other ongoing transmissions causing more packet losses. Therefore, any improvement in this process will automatically benefit the throughput significantly. In case of UDP, this is not an issue. If there is a failure, there is no re-transmission effort and thus interference with other route transmissions is not possible. Therefore, the impact of the new routing metrics on this protocol will be comparably lower.

Finally, comparing the behavior under different propagation models, we see that ITU-R1411 does not benefit from the proposed metrics as much as the Friis model when TCP is considered. We speculate that this might be due to the way these models work. Friis propagation loss model is a simple model in the sense that the only geographical input it takes is the separation between nodes while ITU-R1411 also takes into consideration the height from the ground to calculate the propagation loss. Lower stations experience more attenuation than the ones located at a higher altitude which might be overall benefiting percentage of improvement in Friis more due to all nodes being treated in the same manner.

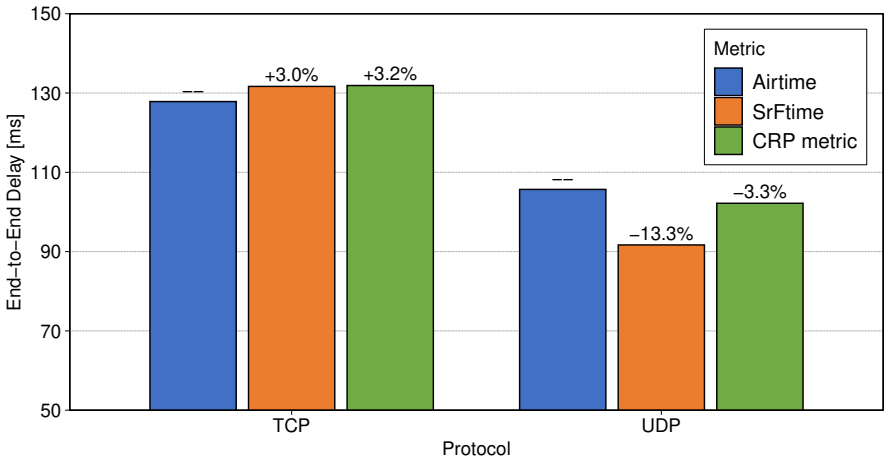
8.3.2 End-to-End Delay Results - Stationary

We next assessed the impact of our proposed metrics on average packet delay. This was needed as increased throughput might increase traffic and hence cause delay.

The results shown in Fig. 8.3 indicate that *SrFTime* also positively contributes to end-to-end delay for both propagation loss models. Specifically, it reduces significantly the overall delay when UDP is used. For TCP, *SrFTime* performs similar to the *Airtime* metric and the trade-off introduced is minimum. The reason behind the delay reduction for UDP could be due to the fact that the proposed metric finds



(a) Friis Propagation Loss Model



(b) ITU-R1411 Propagation Loss Model

Figure 8.3: End-to-End Delay - Stationary FANET

better routes and thus reduces packet delays. In the case of TCP however, this is not apparent as the increased quality in the path only increases reliability and thus throughput but this comes with more overhead in delay or perhaps longer paths which eventually slightly increases delays overall.

The slight decrease in the performance of *CRP* with respect to *SrFTime* follows from the behavior explained when analyzing the throughput performance. By influencing the link metric so that the routing protocol avoids the links very close

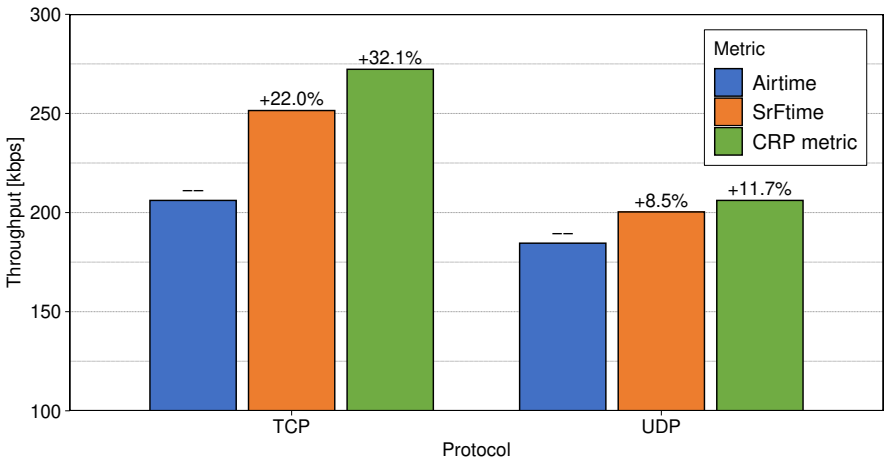
to the coverage limit, we pick the links with medium and shorter distances with *CRP*. Thus, this may potentially increase the number of hops hence increasing the end-to-end delay. These results are expected since stationary nodes will not have broken links for out-of-range mobility issues.

Comparing the different models, the impact on ITU-R1411 is a bit less. Especially for UDP, *Airtime* already provides less delay compared to the one resulting from using Friis propagation model and though there is less room for improvement, there is still a positive impact nonetheless.

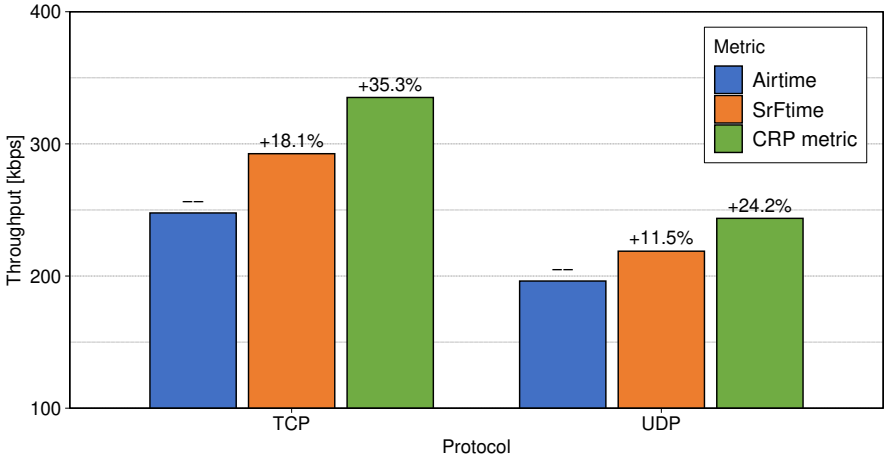
8.3.3 Throughput Results - Mobile Scenarios

In the second part of the simulation experiments, we evaluated the performance of the proposed metrics under mobile topologies. As mentioned, our proposed mobility model has been used in these experiments. The results correspond to the average of testing at different station data rates 5/10/20/40 kbps and across five mobility scenarios.

Based on simulations, we found that the selection of the most optimal weighting coefficient γ for *CRP* metric defined in Equation 5.1 varies when using different propagation loss models. When using Friis propagation loss model, it was determined that $\gamma = 30$ provides a sustained improvement when applied to the different mobility scenarios. Similarly $\gamma = 54$, makes this metric perform better when ITU-R1411 is used. A reason for this is that the 3dB margin considered in that metric can be translated to different distances for different propagation loss models. Then by adjusting the coefficient γ, we can account for those differences and optimize the metric. The results for throughput are shown in Fig. 8.4.



(a) Friis Propagation Loss Model

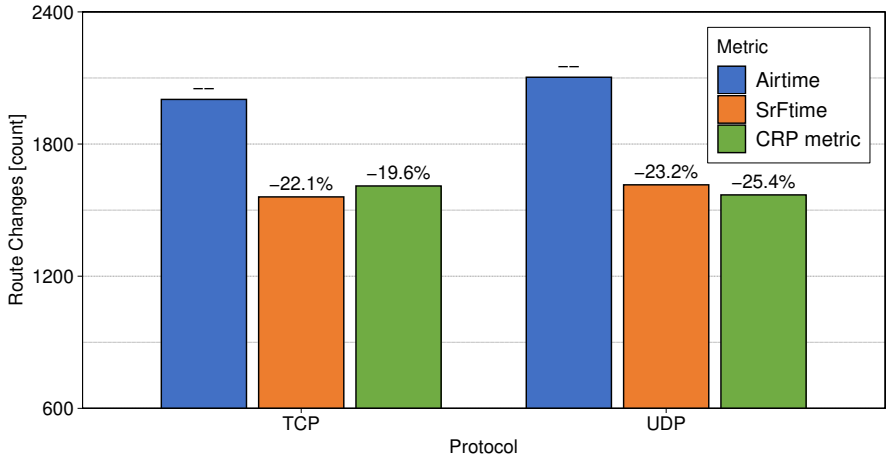


(b) ITU-R1411 Propagation Loss Model

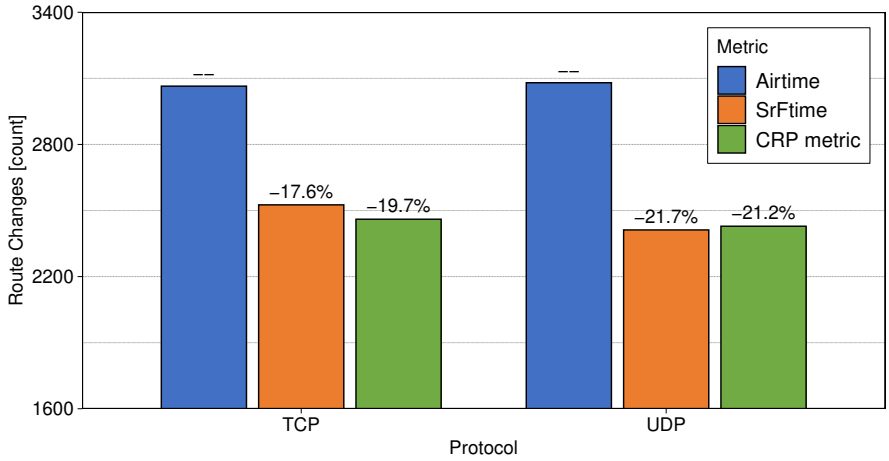
Figure 8.4: Network Throughput in Mobile Scenarios

As seen in the figure, *CRP* metric provides a remarkable increase of over 32% in network throughput compared to *Airtime* for TCP traffic under both propagation loss models. Even for UDP traffic, the improvement is still significant compared to stationary case: 12% when using Friis and 24% when using ITU-R1411 propagation loss model. *SrFTime* still outperforms *Airtime* (i.e., around 20%) but is definitely not as effective as the *CRP* metric. As in the stationary experiments, the number of route changes for both metrics is always less than that of *Airtime* metrics in

any case as seen in Fig. 8.5. We observe that there is slight reduction in *CRP* metric route changes compared to *SrFTime* (which is contrary to stationary case) and thus this helps keeping routes more stable with *CRP* metric, helping to improve the throughput.



(a) Friis Propagation Loss Model



(b) ITU-R1411 Propagation Loss Model

Figure 8.5: Number of Route Changes in Mobile Scenarios

In addition to route stability, these performance measurements corroborate the logic behind the metric definition: making nodes reconsider an alternate next hop node when the current node is very likely to be off-range thus resulting in a link

break and a subsequent re-transmission of TCP packets, proves to be advantageous. When an alternate route is chosen before a link break occurs, a smooth route update takes place, reducing the negative effects of having a link break such as increased buffering and re-transmissions. In the case of UDP there is no re-transmissions, therefore the improvement is slight which can be attributed to reducing the number of packets that would have been lost should an alternate route not be chosen before a link break occurs.

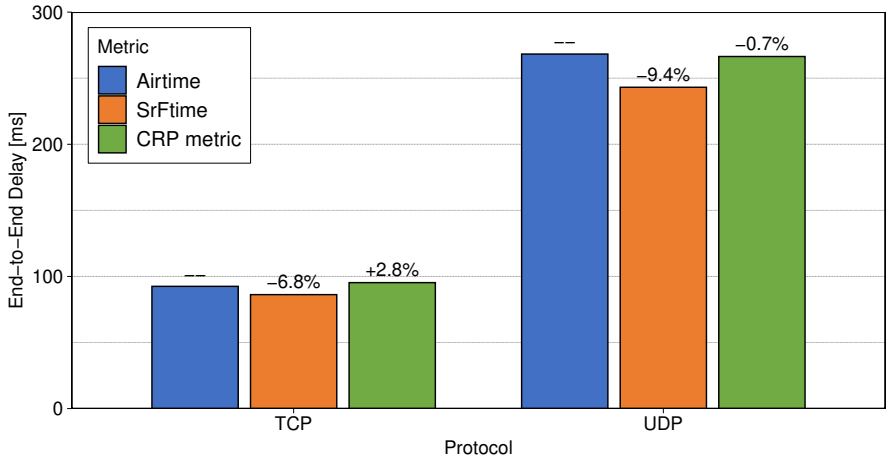
These results also show another interesting trend: Under ITU-R1411 model, *CRP* metric improvement is much better. This was not the case for Friis model. This shows that our metric would be very suitable to be implemented in real environments.

Overall, we can argue that our approach to metric optimization does not aim to counter completely the negative effects of link breaks, but rather provides an overall improvement without increasing the network overhead that occurs when additional probe or control packets are sent.

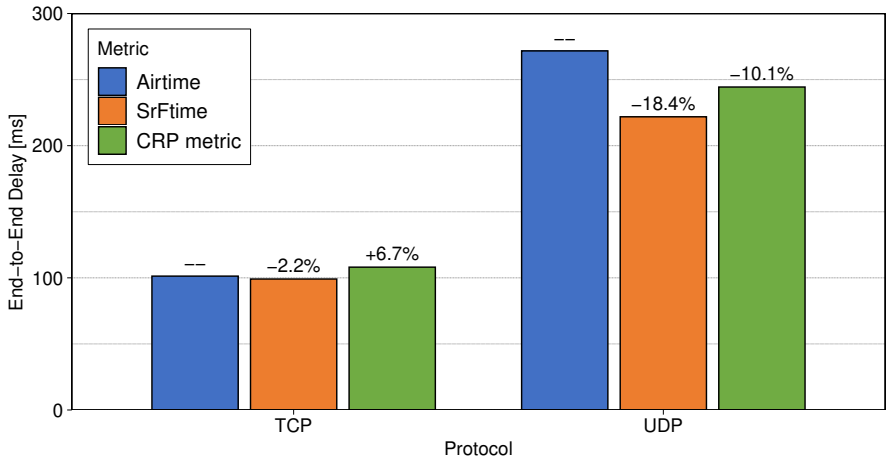
8.3.4 End-to-End delay results for Mobile Scenarios

Next, we analyzed the impact of using our proposed metrics on end-to-end delay under the mobility scenario. The results are shown in Fig. 8.6. From these results, we observe that the results for both *SrFTime* and *CRP* metrics related to *Airtime* for UDP are similar to those obtained in stationary networks and thus there is no impact on delay. When considering the delay under TCP, the results are even better than the stationary case for both metrics. There is reduction in case of *SrFTime* metric while the amount of increase for *CRP* metric is slightly less. It is also interesting that the delay is not impacted from propagation models, again

predicting the practicality of our metrics in real deployments. Overall, there is negligible impact on delay for both metrics under any conditions, indicating the promise for our metrics.



(a) Friis Propagation Loss Model



(b) ITU-R1411 Propagation Loss Model

Figure 8.6: End-to-End delay in Mobile Scenarios

CHAPTER 9

CONCLUSION AND FUTURE WORK

In this research work, we targeted the current standard used for multi-hop communication among FANETs and aimed to improve the network performance through modifying the *Airtime* routing metric of 802.11s standard based on the needs of drone applications, avoiding to increase network overhead by using information available at each node. The new metric included more contribution from the error rates that might fluctuate due to changing environment of drones and their mobility.

Based on the simulation results, we show that the proposed *CRP* metric provides the best performance in terms of network throughput and also a better end-to-end delay. Therefore it can serve the needs of mobile FANETs or swarm-of-drones. *SrFTime* on the other hand is more suited for stationary FANETs which provides a better balance between network throughput and end-to-end delay.

The source code containing the expanded model including both novel routing metrics is freely available for its use [Bau20].

Future work include combining the use of improved mobility models with Tx power control in addition to explore the possibility of expanding the routing table to store an alternate path which will be ready to be used when the preferred path is broken.

BIBLIOGRAPHY

[80211] Ieee standard for information technology–telecommunications and information exchange between systems–local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 10: Mesh networking. *IEEE Std 802.11s-2011 (Amendment to IEEE Std 802.11-2007 as amended by IEEE 802.11k-2008, IEEE 802.11r-2008, IEEE 802.11y-2008, IEEE 802.11w-2009, IEEE 802.11n-2009, IEEE 802.11p-2010, IEEE 802.11z-2010, IEEE 802.11v-2011, and IEEE 802.11u-2011)*, pages 1–372, 10 2011.

[AB11] Kirill Andreev and Pavel Boyko. Ieee 802.11s mesh networking ns-3 model. *IITP, WNS3, March*, 2011.

[ABB10] Riduan M Abid, Taha Benbrahim, and Saad Biaz. Ieee 802.11s wireless mesh networks for last-mile internet access: An open-source real-world indoor testbed implementation. *Wireless Sensor Network*, 2(10):725, 2010.

[AEGPS10] Nils Aschenbruck, Raphael Ernst, Elmar Gerhards-Padilla, and Matthias Schwamborn. Bonnmotion: A mobility scenario generation and analysis tool. In *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, SIMUTools '10, Brussels, BEL, 2010. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[AHR04] Baruch Awerbuch, David Holmer, and Herbert Rubens. High throughput route selection in multi-rate ad hoc wireless networks. In Roberto Battiti, Marco Conti, and Renato Lo Cigno, editors, *Wireless On-Demand Network Systems*, pages 253–270, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[Bau19a] O. Bautista. Ns2mobilityhelper class upgrade to support full 3d mobility in ns-3 network simulator, 2019. [online] Accessed January-2020. https://github.com/ogbautista/Ns2MobilityHelper.

[Bau19b] O. Bautista. Visualization tool for mobility scenarios in bonnmotion format and tools for modification and conversion to ns2 movements format, 2019. [online] Accessed January-2020. https://github.com/ogbautista/Visual_Bm_Scenario".

[Bau20] O. Bautista. Ns3 implementation of srftime and crp routing metrics for ieee 802.11s mesh standard, 2020. [online] Accessed February-2020. https://github.com/ogbautista/RoutingMetricsIeee802-11s.

[BCC+17] A. Bujari, C. Calafate, J.C. Cano, P. Manzoni, C.E. Palazzi, and D. Ronzani. Flying ad-hoc network application scenarios and mobility models. *International Journal of Distributed Sensor Networks*, 13, 10 2017.

[CJ03] T. Clausen and P. Jacquet. Rfc-3626: Optimized link state routing protocol (olsr), Oct 2003.

[CK08] J. D. Camp and E. W. Knightly. The ieee 802.11s extended service set mesh networking standard. *IEEE Communications Magazine*, 46(8):120–126, August 2008.

[DPZ04] Richard Draves, Jitendra Padhye, and Brian Zill. Comparison of routing metrics for static multi-hop wireless networks. In *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '04, page 133–144, New York, NY, USA, 2004. Association for Computing Machinery.

[Fri46] H. T. Friis. A note on a simple transmission formula. *Proceedings of the IRE*, 34(5):254–256, May 1946.

[GC18] A. Guillen-Perez and M. Cano. Flying ad hoc networks: A new domain for network communications. *Sensors (Basel)*, 18(10), October 2018.

[Goo20] Google. Google wifi mesh router, 2020. [online] Accessed February-2020. https://store.google.com/us/product/nest_wifi.

[HGPC99] Xiaoyan Hong, Mario Gerla, Guangyu Pei, and Ching-Chuan Chiang. A group mobility model for ad hoc wireless networks. In *Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '99, page 53–60, New York, NY, USA, 1999. Association for Computing Machinery.

[HYM16] S. Hayat, E. Yanmaz, and R. Muzaffar. Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint. *IEEE Communications Surveys Tutorials*, 18(4):2624–2661, Fourthquarter 2016.

[IBScT13] İlker Bekmezci, Ozgur Koray Sahingoz, and Şamil Temel. Flying ad-hoc networks (fanets): A survey. *Ad Hoc Networks*, 11(3):1254 – 1270, 2013.

[Int17] International Telecommunication Union. *Rec. ITU-R P.1411-9: Propagation data and prediction methods for the planning of short-range outdoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 100 GHz*, Jun. 2017.

[JMB07] D. Johnson, D. Maltz, and J. Broch. Rfc-4728: The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4, Feb 2007.

[KGBV17] C. J. Katila, A. Di Gianni, C. Buratti, and R. Verdone. Routing protocols for video surveillance drones in ieee 802.11s wireless mesh networks. In *2017 European Conference on Networks and Communications (EuCNC)*, pages 1–5, June 2017.

[KRN10] G Vijaya Kumar, Y Vasudeva Reddyr, and Dr M Nagendra. Current research work on routing protocols for manet: a literature survey. *international Journal on computer Science and Engineering*, 2(03):706–713, 2010.

[LGZ+19] N. Lin, F. Gao, L. Zhao, A. Al-Dubai, and Z. Tan. A 3d smooth random walk mobility model for fanets. In *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 460–467, Aug 2019.

[LLK18] G. A. Litvinov, A. V. Leonov, and D. A. Korneev. Applying static mobility model in relaying network organization in mini-uavs based fanet. In *2018Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*, pages 1–7, July 2018.

[LW07] Fan Li and Yu Wang. Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular technology magazine*, 2(2):12–22, 2007.

[MD01] M. K. Marina and S. R. Das. On-demand multipath distance vector routing in ad hoc networks. In *Proceedings Ninth International Conference on Network Protocols. ICNP 2001*, pages 14–23, Nov 2001.

[MSS13] G. Mohandas, S. Silas, and S. Sam. Survey on routing protocols on mobile adhoc networks. In *2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)*, pages 514–517, March 2013.

[Nay18] A. Nayyar. Flying adhoc network (fanets): Simulation based performance comparison of routing protocols: Aodv, dsdv, dsr, olsr, aomdv and hwmp. In *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, pages 1–9, Aug 2018.

[ns-18] Ns-3, a discrete event network simulator, Sep. 2018. [online] https://www.nsnam.org/.

[oB16] University of Bonn. Bonnmotion: A mobility scenario generation and analysis tool, 2016. [Online] Accessed December-2019. https://sys.cs.uos.de/bonnmotion/.

[PB94] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, SIGCOMM '94, page 234–244, New York, NY, USA, 1994. Association for Computing Machinery.

[PBRD03] C. Perkins, E. Belding-Royer, and S. Das. Rfc-3561: Ad hoc on-demand distance vector (aodv) routing, Jul. 2003.

[PKB+09] Georgios Parissidis, Merkourios Karaliopoulos, Rainer Baumann, Thrasyvoulos Spyropoulos, and Bernhard Plattner. *Routing Metrics for Wireless Mesh Networks*, pages 199–230. Springer, London, 02 2009.

[PZ18] Paszkiewicz, Andrzej and Zapala, Przemyslaw. The modified metric for self-organization wireless mesh networks. *ITM Web Conf.*, 21:00010, 2018.

[QBN+19] Vu Khanh Quy, Nguyen Tien Ban, Vi Hoai Nam, Dao Minh Tuan, and Nguyen Dinh Han. Survey of recent routing metrics and protocols for mobile ad-hoc networks. *Journal of Communications*, 14(2):110–120, 2019.

[SAAU18] N. Saputro, K. Akkaya, R. Algin, and S. Uluagac. Drone-assisted multi-purpose roadside units for intelligent transportation systems. In *2018*

IEEE 88th Vehicular Technology Conference (VTC-Fall), pages 1–5, Aug 2018.

[SAI+18] M. K. Singh, S. I. Amin, S. A. Imam, V. K. Sachan, and A. Choudhary. A survey of wireless sensor network and its types. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pages 326–330, Oct 2018.

[SAU18] N. Saputro, K. Akkaya, and S. Uluagac. Supporting seamless connectivity in drone-assisted intelligent transportation systems. In *2018 IEEE 43rd Conference on Local Computer Networks Workshops (LCN Workshops)*, pages 110–116, Oct 2018.

[ZXG] Biao Zhou, Kaixin Xu, and Mario Gerla. Group and swarm mobility models for ad hoc network scenarios using virtual tracks. In *IEEE MILCOM 2004. Military Communications Conference, 2004.*, volume 1, pages 289–294. IEEE.